

This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (Currently Amended) A system comprising:
a processor ~~configured~~ for executing computer-executable instructions; and
a computer-readable storage medium having encoded thereon computer-executable instructions to support the enforcement of a license for a computer program subject to use under a plurality of licenses each permitting different rights in the computer program, the ~~processor~~ computer-readable storage medium comprising:
 - a licensing component common to the plurality of licenses for the computer program, the licensing component maintains a license store in which the licenses are stored, the licensing component further maintains a trust store in which dynamic data is stored in a tamper-resistant manner, wherein the dynamic data is utilizable to validate the licenses, each license comprising at least one right in the computer program and a set of data associated with said at least one right, the licensing component exposing a callable interface to the computer program, said callable interface comprising:
 - a right-consumption component which receives an identifier of a right from the computer program and determines whether the right can be exercised; and
 - an information-retrieval component which receives an identifier of said right from the computer program and provides said set of data, or information based on said set of data, to the computer program.
2. (Previously Presented) The system of claim 1, wherein said licensing component is common to and usable by a plurality of different computer programs and said license store stores licenses for the plurality of different computer programs, the computer program being included among said plurality of different computer programs, wherein said callable interface further comprises:
 - a handle-opening component that provides a handle to the computer program;

wherein the rights-consumption component receives the handle from the computer program and uses the handle to identify the computer program from which a call to the rights-consumption component is received; and wherein the computer program and licensing component are located on a single computing device.

3. (Previously Presented) The system of claim 1, wherein the rights-consumption component causes the licensing component to select a license based on one or more factors comprising:

whether the license store is associated with the computer program; and
a conflict rule that determines which license to select from among a plurality of licenses that are associated with the computer program.

4. (Previously Presented) The system of claim 1, wherein the licensing component does not enforce licensing constraints on the computer program, and wherein said callable interface further comprises:

an asynchronous-context-initiator component that establishes a context for asynchronous processing and provides an identifier of said context to the computer program;

wherein said rights-consumption component receives the identifier of said context from said computer program and processes a right-consumption request asynchronously in response to receipt of the identifier of said context.

5. (Previously Presented) The system of claim 1, wherein the rights-consumption component determines whether the right can be exercised based on whether the right is identified in a license.

6. (Previously Presented) The system of claim 1, wherein the computer program and the licensing component execute on a machine, and wherein the rights-consumption component

determines whether the right can be exercised based on whether the license is bound to said machine.

7. (Currently Amended) The system of claim [[1]] 6, wherein the computer program is associated with a product identifier, and wherein the rights-consumption component determines whether the right can be exercised based on whether the license is bound to said machine or to a class of machines of which said machine is a member.

8. (Currently Amended) A method, implemented by at least one machine, of restricting the use of a computer program associated with a plurality of licenses, each license specifying at least one right in the computer program, the method comprising:

invoking, via the machine, a licensing service common to the plurality of licenses by making a first call from the computer program to a first method of an interface of said licensing service, the licensing service in communication with a trust store in which dynamic data is stored in a tamper-resistant manner, wherein the dynamic data is utilizable to validate licenses, said first call being parameterized by an identifier associated with a right;

in response to said first call receiving an indication as to whether the right is exercisable; and

engaging in a first behavior or a second behavior according to the indication.

9. (Original) The method of claim 8, wherein said first behavior comprises allowing the computer program to execute, and wherein said second behavior comprises discontinuing execution of the computer program.

10. (Original) The method of claim 8, wherein said first behavior comprises allowing the computer program to perform a first set of functions, and wherein said second behavior

comprises allowing the computer program to perform a second set of functions that is non-identical to said first set of functions.

11. (Original) The method of claim 8, wherein the right is associated with a set of data, wherein the method further comprises:

making a second call to a second method of said interface, said second method being parameterized by an indication of the right; and
in response to said second call, receiving said set of data.

12. (Original) The method of claim 11, further comprising:

directing the operation of the computer program based on said set of data.

13. (Original) The method of claim 8, further comprising:

making a second call to a second method of said interface; and
in response to said second call, receiving a handle;
wherein said second call is made prior to said first call, and wherein said first call is further parameterized by said handle.

14. (Original) The method of claim 8, further comprising:

making a second call to a second method of said interface; and
in response to said second call, receiving an asynchronous context;
wherein said second call is made prior to said first call, wherein said first call is further parameterized by said asynchronous context, and wherein the computer program performs at least one action while the first call is handled asynchronously.

15. (Original) The method of claim 8, wherein said first method determines whether the right is exercisable based on one or more factors comprising:

whether the license is bound to a machine or environment on which the computer program is executing;

whether the license or right is bound to a product identifier associated with the computer program;

whether the license or right has expired; and

whether the right has been consumed a number of times in excess of a right specified in the license.

16. (Previously Presented) A computer-readable medium having encoded thereon computer-executable instructions to perform a method of enabling the enforcement of a license to a computer program that is subject to use under a plurality of licenses each permitting different rights in the computer program, the method comprising:

receiving a first method call from the computer program, the first method call identifying a right in the computer program;

determining that the right is contained in a particular license among the plurality of licenses and is exercisable through the use of a license store and a trust store having stored therein dynamic data that is stored in a tamper-resistant manner and is utilizable to validate the plurality of licenses; and

returning to the computer program an indication that the right is exercisable.

17. (Previously Presented) The computer-readable medium of claim 16, wherein the indication comprises a binding of the right to the particular license.

18. (Previously Presented) The computer-readable medium of claim 16, wherein said determining act is based on whether the right is specified in the particular license.

19. (Previously Presented) The computer-readable medium of claim 16, wherein said determining act is based on whether the particular license is bound to a machine on which the computer program is executing.

20. (Previously Presented) The computer-readable medium of claim 16, wherein said determining act is based on whether the particular license or the right is bound to the computer program.

21. (Previously Presented) The computer-readable medium of claim 16, wherein said determining act is based on whether the particular license or the right is non-expired.

22. (Previously Presented) The computer-readable medium of claim 16, wherein said determining act is based on whether the particular license has been consumed a number of times that exceeds a limit.

23. (Original) The computer-readable medium of claim 16, wherein the method further comprises:

- receiving a second method call from the computer program; and
- in response to the second method call, returning a handle to the computer program that identifies the computer program;

wherein said first method call is performed subsequent to said second method call, and wherein said first method call further identifies said handle.

24. (Original) The computer-readable medium of claim 16, wherein the method further comprises:

- receiving a second method call from the computer program;

in response to the second method call, returning an asynchronous context to the computer program, wherein the first method call is executed subsequent to the second method call and identifies said asynchronous context; and
executing the first method call asynchronously while the computer program performs an action.

25. (Original) The computer-readable medium of claim 16, wherein the right is associated with a set of data, and wherein the method further comprises:
receiving a second method call which indicates the right; and
in response to said second method call, providing the set of data to the computer program.